Project Proposal - LLM Generated Text Detection

Abde Manaaf Ghadiali - G29583342, Gehna Ahuja - G35741419, Sagar Sheth - G32921700, Venkatesh Shanmugam - G27887303

Project Objective

After the advent of Large Language Models (LLM), we observed an enhanced efficiency of frameworks of various industries which adopted LLMs. However, we see another trend which talks about the misuse of LLMs. For example, students could use LLMs to generate essays to complete their assignments which is AI-Plagiarism and that could impact students' skill development which is a major concern of educators these days. Therefore, our aim is to develop a framework that can distinguish between human-written and LLM-generated text.

The framework will be an ensemble of various Machine Learning and Deep Learning algorithms, rather than solely relying on a single model, and that is what sets us apart from the available approaches. The dataset will also be a unique combination of texts generated on diverse topics through LLMs and human-written essays. Additionally, we will try to employ Explainable Techniques to provide insights into the decision-making process of our model.

Data Sources

The data for this project will be gathered from various sources, including Kaggle, HuggingFace, GitHub, etc. Additionally, we will utilize Large Language Models (LLMs) such as ChatGPT, Gemini, etc. to generate text on specific topics, contributing to the corpus. To ensure diversity, we will extract text from different sources on the web that we know are written by humans and not AI. Below is the list of data sources used in this project:

- Human vs LLM Text Corpus | Kaggle
- DAIGT v3 Train Data (Human and AI) | Kaggle
- LLM - Detect AI Generated Text | Kaggle
- Deepfake Text Detect | Github
- Essay With Instructions | HuggingFace

We also possess a corpus of human-written text that we will utilize as a holdout set or test set for assessing the accuracy of our final model.

Methodology

We are going to create a Binary Classification Model, which will ingest the data collected from the different sources mentioned above and classify whether the text was human-written (class 0) or AI-generated (class 1).

Since the scope of the project is generally extensive, responsibilities will be divided equally among the team members for the tasks mentioned below. All team members will collaboratively work on each task, ensuring that no individual works on any one thing independently. Mentioned below is a summary of the critical as well as supportive tasks we shall undertake during the course of this project.

Critical Tasks (Actual Deliverables):

- Data Source Validation and Data Gathering

- Exploratory Data Analysis (EDA)
- Data Preprocessing and Featurization
- Designing the Architecture of the Model
- Hyperparameter Optimization
- Obtaining and Inferencing the Results (includes Explainable Techniques)
- Final Report for the Project

Supportive Tasks (Better Presentation):

- (**Only if Required**) Combining the computational power of our systems to build a larger distributed system for faster computations.
- Creating a User Interface (UI) on Plotly Dash (Python Framework) for Demonstration (**If Time Permits**).
- Optimization and Refactoring of code with design principles and industry standards.

Success Metrics and Risks Factors

The baseline performance will involve classifying the input text at random, yielding an accuracy of 0.5. Achieving results beyond this threshold will be considered a success. The primary metric for the scope of this project will be the F1 Score, as both False Negatives and False Positives are equally important. Additionally, the correct context for explaining why our model is classifying a given text as AI-generated would be a critical factor for success.

Several potential challenges could impact the success of our project. Firstly, the limitations of our system hardware might affect the efficiency and speed of our model training and scalability. We plan to address this by taking precautions, such as reducing the number of sentences for the input and utilizing the integrated GPU of our systems. If needed, we will leverage the free hardware-accelerated services provided by Google Colab/Kaggle. Another challenge is ensuring that our model generalizes over a broad spectrum of topics rather than a narrow focus. We aim for our model to be unbiased. However, to manage the project scope, we will limit training and testing to a subset of topics derived from our training data, measuring success accordingly.

Initial References

[1] G. Gritsay, A. Grabovoy and Y. Chekhovich, "Automatic Detection of Machine Generated Texts: Need More Tokens," 2022 Ivannikov Memorial Workshop (IVMEM), Moscow, Russian Federation, 2022, pp. 20-26, doi: 10.1109/IVMEM57067.2022.9983964.

[2] Gaggar, Raghav, Ashish Bhagchandani, and Harsh Oza. "Machine-Generated Text Detection using Deep Learning." arXiv preprint arXiv:2311.15425 (2023).

[3] Jawahar, Ganesh, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. "Automatic detection of machine generated text: A critical survey." arXiv preprint arXiv:2011.01314 (2020).

[4] Kadhim Hayawi, Sakib Shahriar, Sujith Samuel Mathew," The Imitation Game: Detecting Human and AI-Generated Texts in the Era of ChatGPT and BARD",doi.org/10.1177/016555152412275.

[5] Wenxiong Liao, Zhengliang Liu, Haixing Dai, Shaochen Xu, Zihao Wu, Yiyang Zhang, Xiaoke Huang, Dajiang Zhu, Hongmin Cai, Tianming Liu, Xiang Li, "Differentiating ChatGPT-Generated and Human-Written Medical Texts: Quantitative Study", doi: 10.2196/48904